Parallel Digital Architectures for High-Speed Adaptive DSSS Receivers

Stephan Berner and Phillip De Leon
New Mexico State University
Klipsch School of Electrical and Computer Engineering
Las Cruces, New Mexico 88003-8001
{sberner, pdeleon}@nmsu.edu

Abstract

DSP-based implementations of receivers have many advantages over their analog counterparts including precise matched filtering and reconfigurability. As processing rates increase, more receiver functions are implemented digitally–the ultimate goal in this shift being all-digital receivers which sample at IF or RF. Practical limitations obviously occur when processing rates fall behind sampling and symbol rates. In this paper, we extend previous ideas for introducing parallelism into the receiver design. We describe a parallel, adaptive DSSS receiver in which individual processing units can potentially operate at rates below the symbol rate. The design is shown to have BERs equivalent to conventional designs.

1. Introduction

The advantages of digital (DSP) implementations of receivers over their analog counterparts are well known: precise matched filtering, temperature stability, possible reductions in physical size and power consumption, reconfigurability for different modulation schemes and data rates, etc.... These digital implementations naturally require the sample rate to be much lower that the processing rate (clock frequency) of the hardware so that appropriate processing can be completed in time. As processing rates increase, higher data rates can be accomodated and more receiver functions can be digitally implemented—the ultimate goal in this shift being all-digital receivers which sample at Intermediate Frequency (IF) or even Radio Frequency (RF). For gigabit receivers, however, the symbol rate (not to mention the sampling rate) exceeds current practical processing rates.

In the mid-1990s, pioneering work at the Jet Propulsion Laboratory (JPL) applied architectural parallelism to all-digital, high-rate BPSK receiver designs to circumvent the limitation of desired receiver data rates being higher than digital processing rates [1]. The architectural parallelism is accomplished through the use of oversampled filterbanks which provide a serial-to-parallel conversion thereby creating multiple, lower-rate signals. The various receiver elements are then designed around these lower-rate signals to accomplish tasks of detection and estimation of multiple symbols per clock cycle. Simulations of the system assumed an IF signal is bandpass-sampled at a rate of $f_s = 100$ MHz and split into 16 subbands with a $2\times$ oversampled, uniform-DFT filterbank (efficiently implemented in polyphase form). The subband sampling rate is therefore reduced to 12.5MHz. Subband equivalents of a digital phase-locked loop (PLL) and combined demodulator and matched filter were also developed. Simulation results indicated no real bit error rate (BER) difference between the serial and parallel implementation thus validating the proposed design.

More recently, these ideas were used in a 1.2Gbps, parallel, 16-QAM receiver implemented on a single, 125MHz, CMOS ASIC [2]. In this work, a 32-subband, $2\times$ oversampled, uniform-DFT filter bank was used in the serial-to-parallel conversion. In order to reduce complexity, a low-order (sub-optimum), subband detection filter was used which had the effect of introducing Inter-Symbol Interference (ISI). However, a novel parallel equalizer was employed to compensate. Simulation results of the parallel 16-QAM receiver indicated a relatively minor 0.5dB loss in BER performance as compared to a serial architecture.

In this paper, we apply previous ideas for introducing parallelism into receiver designs and propose a parallel adaptive direct-sequence spread spectrum (DSSS) receiver. In an adaptive DSSS receiver, instead of a fixed matched filter, the receiver uses an adaptive filter to help overcome interference after training [3]. In addition, there is no pseudo-noise (PN) code aquisition phase since the adaptive filter will contain an approximation of the PN code after training (assuming moderate interference). Our motivation for this study is to develop a parallel architecture for an all-digital, DSSS receiver implementation whose processing

rate (clock speed) can be made significatly less than the symbol rate. We show parallel, adaptive DSSS receivers can be designed which provide similar BERs as compared to conventional, serial designs.

2. Conventional, Serial Adaptive DSSS Receivers

Adaptive DSSS receivers can overcome the degrading effects of multiple-access interference (MAI), narrow band interference, and ISI without having information about interferers or channel. Unlike a receiver with a fixed matched filter, an adaptive receiver consists of an adaptive filter, which cancels undesired signal components after training. Another advantage of the receiver is that it requires no information about the PN code (other than its length) and does not require a code acquisition phase [3].

There are two types of serial adaptive DSSS receivers: chip-spaced (CS) and fractionally-spaced (FS). In the CS adaptive receiver illustrated in Fig. 1, the received signal, $x(n)$ is passed through a chip matched filter and the output sampled at the chip rate, T_c. The resulting chip samples are then passed through an adaptive filter, $\mathbf{w}$ (length spans at least one symbol) and the output sampled at the symbol rate T_s. This output, $y(n)$ is used to estimate the transmitted symbol, $\hat{a}(n)$, and to adjust the adaptive filter at the symbol rate during training. The FS adaptive receiver illustrated in Fig. 2, does not require a chip matched filter. In this system, the received signal, $x(n)$ (sampled at a sufficiently high rate) is passed through an adaptive filter, $\mathbf{w}$ and the output sampled at the symbol rate T_s. As in the CS receiver, this output, $y(n)$ is used to estimate the transmitted symbol and to adjust the adaptive filter.

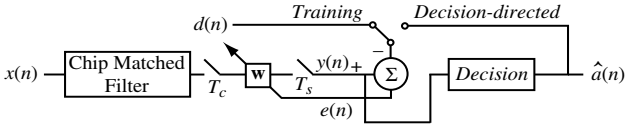

Figure 1. Chip-spaced adaptive DSSS receiver.

The advantage of the CS receiver is a shorter adaptive filter while the advantages of the FS receiver are better interference suppression capabilities and lack of the chip matched filter and necessary chip timing synchronization. The proposed parallel receiver (described in the next section) is based on the FS receiver since parallelization of the chip matched filter and chip timing synchronization would add additional complexity to the design problem. In either CS or FS receiver implementation, the adaptive filter may be implemented in a polyphase form to achieve a level

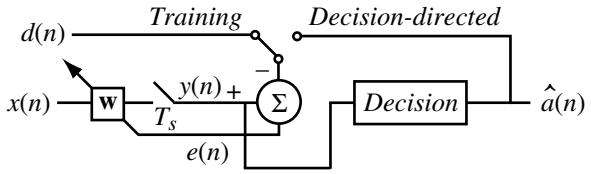

Figure 2. Fractionally-spaced adaptive DSSS receiver.

of parallelism. If we wanted to only introduce parallelism into the receiver, a simple polyphase implementation of the adaptive filter would suffice. However, more general signal processing methods are available to construct such parallel decompositions for adaptive filters [4]. These are explored in the next section.

3. Parallel Adaptive DSSS Receivers

In proposed parallel adaptive DSSS receiver illustrated in Fig. 3, the received signal, $x(n)$ is partitioned into length-M windows and decomposed by a linear transformation, $\mathbf{T}$, into M lower rate (subband) signals, $x_1(n), \ldots, x_M(n)$ [4]. It is assumed that the number of input samples belonging to one symbol is a multiple of the window size M. The lower rate signals are passed through M adaptive filters $\mathbf{w}_1, \ldots, \mathbf{w}_M$ (each shorter than in the serial case and updated at the symbol rate) and the outputs, $y_1, \ldots, y_M$ are sampled at the symbol rate. The resulting signals, $u_1, \ldots, u_M$ are scaled by adaptive gain factors (described below), $\alpha_1, \ldots, \alpha_M$ to yield $\nu_1, \ldots, \nu_M$. These signals are then applied to the inverse transformation, $\mathbf{T}^{-1}$ and the first element picked off to yield z_1. We use z_1 to estimate the transmitted symbol, $\hat{a}(n)$.

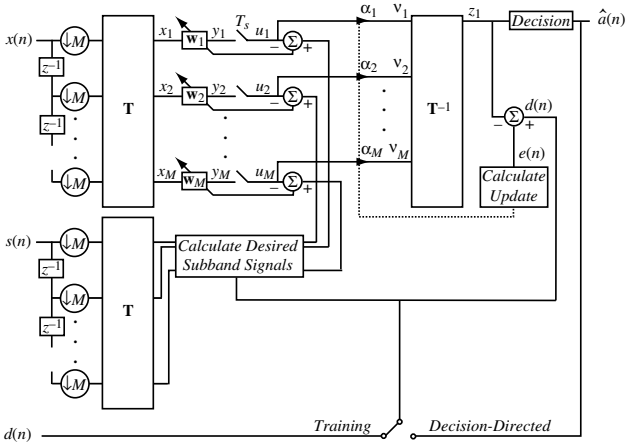

Figure 3. Parallel, adaptive DSSS receiver.

We note that symbol rate sampling in the subbands is equivalent to sampling the output of $\mathbf{T}^{-1}$ (picking off the first element, z_1) due to the fact that the length of one symbol is a multiple of M. Due to the sampling of the output of $\mathbf{T}^{-1}$, much of the computation involved with $\mathbf{T}^{-1}$ can be eliminated since we only need z_1. Furthermore, if the gain factors are initialized with the first row of $\mathbf{T}^{-1}$, the inverse transform block, $\mathbf{T}^{-1}$ in Fig. 3 can be replaced by a summation.

The vector of desired subband signals is obtained in the following way. For the serial receiver, the desired signal is the convolution of the oversampled, pulse-shaped PN sequence [weighted by the symbol (bit)] with the matched filter sampled at the symbol time. Alternatively, this desired signal is the inner product of the oversampled, pulse-shaped PN sequence with itself weighted by the symbol. For the parallel receiver, the vector of desired subband signals is the inner product of the subband version of the PN sequence [$s(n)$ transformed by $\mathbf{T}$] with itself weighted by the bit, $d(n)$.

The purpose of the adaptive gain factors is to speed up the convergence and are considered just another adaptive filter which takes the outputs of the subband filters and weights them in order to minimize the error, $e(n)$. Due to their length, the gain factors can adapt much faster as compared to the subband filters which are typically longer. Therefore during training, the gain factors adapt quickly in order to maintain the lowest possible error until the subband filters have converged. The gain factors, however, cannot lead to the lowest possible Mean Squared Error (MSE) (and BER) because the subband filters have more degrees of freedom to minimize the error, but they can reduce the error during training enough such that the decision-directed mode can be applied earlier. The update of the subband filters and the gain factors are independent of each other and can be viewed as two cascaded control loops.

The choice of the transform, $\mathbf{T}$ greatly effects the performance of the receiver. A good transform should minimize the output MSE as well as the eigenvalue spread of the subband autocorrelation matrices. This will ensure relatively fast convergence for the LMS adaptive filters. Simulation results using several standard transforms (DCT, Hadamard, and DFT) are given in the next section. The search for better transformations is currently being investigated.

4. Simulation Results

BER performance of the parallel adaptive DSSS receiver was simulated and compared to theory, the matched filter (MF) receiver, and the serial adaptive DSSS receiver. System parameters include a length 31 PN sequence, chip pulses shaped with a square-root raised cosine (SRRC) filter (50% excess bandwidth), and a NLMS adaptive algorithm; we assume perfect carrier and chip synchronization. For each simulation point 100,000 symbols are used. In addition, we assume four other users and four narrowband interferences (sinusoids), each 6dB stronger than the desired signal; zero mean, white Gaussian noise is also added to achieve a desired SNR which during training is 6dB. In the serial receiver, the adaptive filter length is 128 (length 31 PN sequence, 4 samples per chip which yields 124 samples, resampled to get 128 samples). In the parallel receiver, we use $M = 8$ subbands with a subband adaptive filter length of 16.

Comparison of convergence times of the serial and parallel adaptive receivers is based on the same BER performance, e.g. the BER of the serial receiver was adjusted by varying the step size of the update algorithm until it was approximately the same as for the parallel receiver. Then convergence time was measured by how long it takes the MSE to come within 10% of the steady-state value. The BER is equivalent to the MSE after convergence for a white, Gaussian error signal (usual assumption) [3]. The BER, however, is not equivalent to the misadjustment because the minimum error resulting from the Wiener solution for the parallel receiver is in general different from the minimum error for the serial receiver [3]. Figs. 4 and 5 show the BER performance and learning curve for the serial receiver. Figs. 6 and 7 show the BER performance and learning curve for the parallel receiver (Hadamard transform) without gain factors. For this parallel receiver, there is little improvement in convergence time over the serial case. Fig. 8 shows the learning curve of the parallel receiver (Hadamard transform) with adaptive gain factors. It can be seen that the error is reduced rapidly during the first 100 symbols of training due to the adaption of the gain factors. The performance of the parallel receiver with different transformations is summarized in Table 1. Receivers built around the DCT and Hadamard transforms performed well while those with the DFT performed poorly.

We also conducted simulations using the RLS algorithm. In this case the convergence time in symbols is roughly equal to the filter length using the same criteria of reducing MSE to below 10% as in Table 1. Therefore the parallel receiver adapts M times faster as compared to the serial receiver. Complete convergence takes longer (serial, 500 symbols; parallel, 150 symbols). Furthermore in the case of RLS the parallel case is computationally more efficient because M RLS updates of short filters is less costly than updating a single, longer filter.

5. Conclusions

Parallel architectures for all-digital receivers enable high data rates using multiple, lower-rate processing units while enjoying many advantages due to digital implementation.

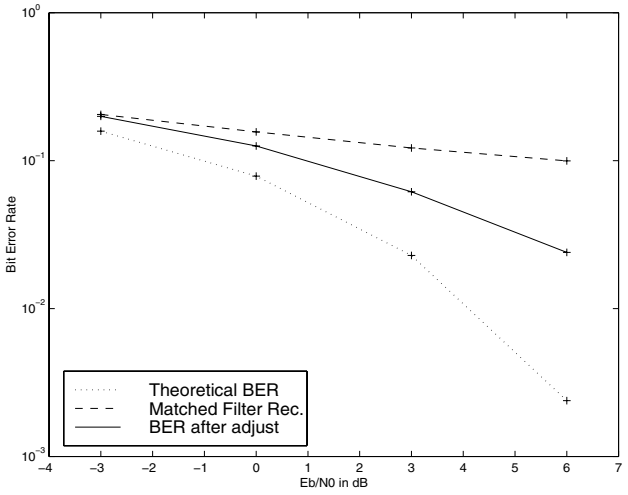

Figure 4. Bit error rate for the serial receiver.

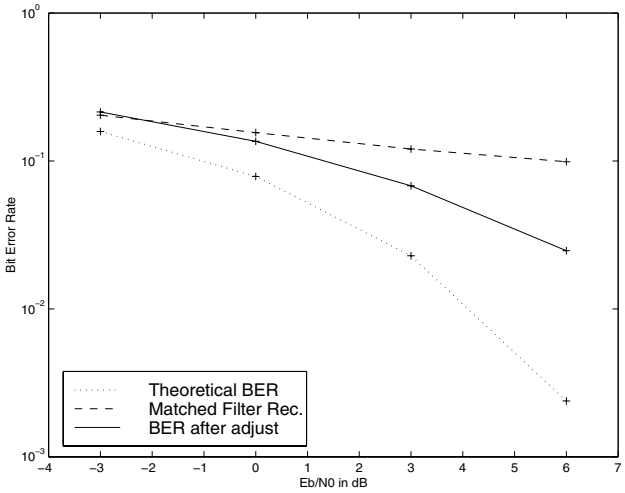

Figure 6. Bit error rate for the parallel receiver (Hadamard transform).

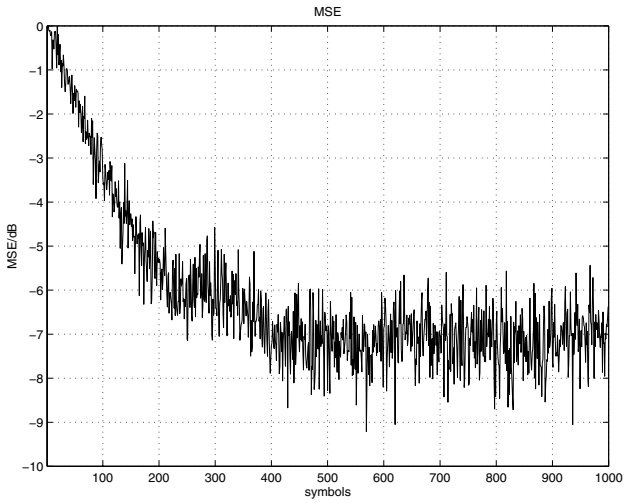

Figure 5. Learning curve for the serial receiver.

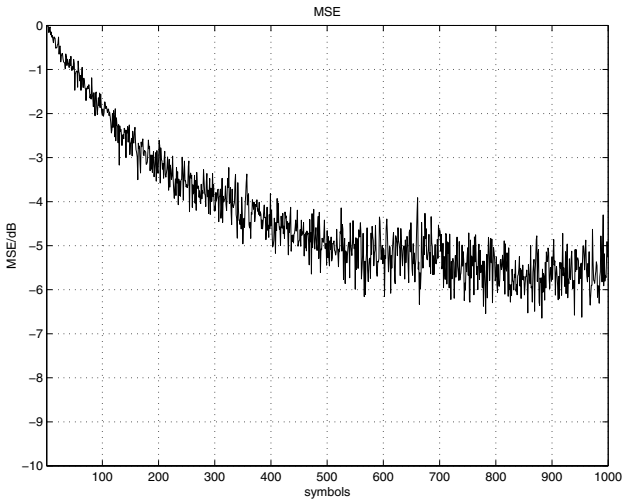

Figure 7. Learning curve for the parallel receiver without gain factors (Hadamard transform).

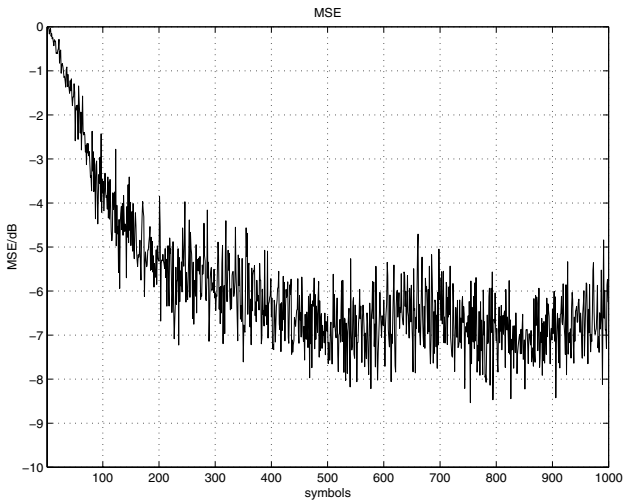

Figure 8. Learning curve for the parallel receiver with adaptive gain factors (Hadamard transform).

Table 1. Performance of Serial and Parallel (Transformed) Adaptive DSSS Receivers.

Transform, **T**	BER for Various SNRs			Conv. Time MSE_{∞}+10%
	0dB	3dB	6dB	
Theoretical	0.0880	0.027	0.0030	N/A
MF Receiver	0.1558	0.1210	0.0999	N/A
Serial Adapt.	0.1125	0.0571	0.0221	400 symbols
DCT	0.1325	0.0656	0.0252	350
DCT w/ Gain Facts	0.1325	0.656	0.0252	150
Hadamard	0.1359	0.0706	0.0288	400
Hadamard w/ Gain Facts	0.1356	0.0697	0.0281	200
DFT	No Improvement Over Matched Filter Receiver			
DFT w/ Cross Facts				

In this paper, we have applied previous ideas on parallel QAM receivers by presenting an architecture for an adaptive DSSS receiver. This parallel receiver, when utilizing DCT or Hadamard transforms and adaptive gain factors, has a faster convergence than the conventional, serial receiver for a fixed a BER.

Acknowledgment

The authors wish to acknowledge the support of this research by NASA, Grant #NAG 5-9323.

References

[1] R. Sadr, P. P. Vaidyanathan, D. Raphaeli, and S. Hinedi, "Parallel digital modem using multirate digital filter banks," *JPL Publication 94-20*, Aug. 1994.

[2] A. Gray, E. Satorius, and P. Ghuman, "All-digital architectures for implementation of a very high data rate 16-qam demodulator and equalizer," *Proc. Int. Conf. Sig. Proc. App. & Tech. (ICSPAT)*, 1999.

[3] S. Miller, "An adaptive direct-sequence code-division multiple-access receiver for multiuser interference rejection," *IEEE Trans. Comm.*, pp. 1746–1754, Feb./Mar./Apr. 1995.

[4] M. Petraglia and S. Mitra, "Adaptive fir filter structure based on the generalized subband decomposition of fir filters," *IEEE Trans. Circuits Sys. II*, vol. 40, pp. 354–362, Jun. 1993.